Ｗｉｎｄｏｗｓによる Ｉ／Ｏ制御技術修得

第３技術室システム設計技術班　　松山　幸雄

１．はじめに

以前パソコンに使用されていたＯＳはＭＳ－ＤＯＳでシングルジョブ，ＣＵＩ（Character User Interface）での文字ベースによる操作であった。近年，ＣＰＵの高速化，メモリの大容量化により，ＯＳはマルチジョブ対応のＷｉｎｄｏｗｓが出現し，ＧＵＩ(Graphical User Interface)でのグラフイックよる簡単な操作でパソコンを扱えるようになった。

Ｗｉｎｄｏｗｓ環境でのアプリケーションプログラム開発言語として，Visual Basic（ＶＢ），Visual C＋＋(ＶＣ＋＋)などがあるが，本研修では，比較的容易にプログラムを作成できるＶＢを使用して，Ｗｉｎｄｏｗｓ上からパソコンに装備されているプリンタポート（パラレルポート）を利用してパラレル通信で制御する調光器のインタフェースとプログラム，RS232Cポート（シリアルポート)を利用してシリアル通信で計測機器制御と波形取得のプログラムを制作したので報告する。

２．プログラム開発環境

図１に示すようにＷｉｎｄｏｗｓでＩ/Ｏを制御する場合，ハードウェアとしてパソコンに標準装備されているシリアル，パラレルポートのレガシィインタフェースや，外部汎用パラレルインターフェースを介して行い，開発言語は主にＶＢ，ＶＣ＋＋を使用するが，ＶＢにはＩ/Ｏ制御するコマンドや関数が無く，ActiveＸコントロール，ＡＰＩ(Application Program Interface)，ＤＬＬ(Dynamic Link Library)を使用して行う。ＡＰＩはWindows上で実行するアプリケーションで使う関数やメッセージで構成され，システムに最初から組み込まれている関数ライブラリである。ＤＬＬはＣ，Ｃ＋＋で開発されＶＢの関数やＡＰＩにない機能やアプリケーションの実行速度を速めたりするのに有効で，複数のアプリケーションがから共有できる共有ファイルとなっている。ActiveＸコントロールは既に作成したフォームなどを再利用するＧＵＩ部品の総称で，ＶＢのランタイムライブラリに含まれ起動時にツールボックスに表示される「標準コントロール」と，独立したファイル(OCXファイル)で必要に応じて後からツールボックスに追加する「カスタムコントロール」がある。本研修では，パラレル通信での制御プログログラムに文献付録[1]のＤＬＬ(DIO32S.DLL)を使用した。シリアル通信での制御プログログラムに

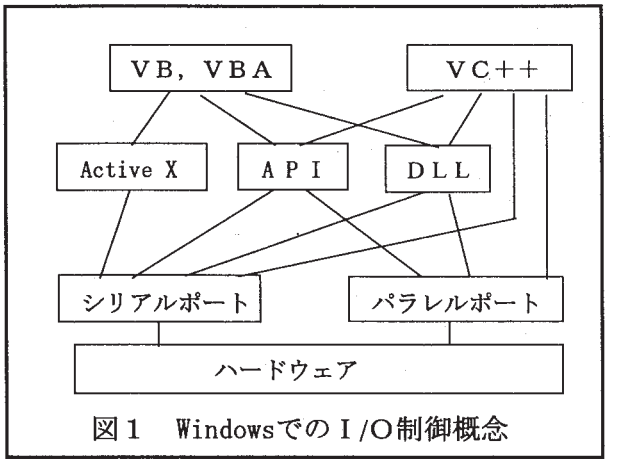

図１　WindowsでのＩ/Ｏ制御概念

は，ＡＰＩやActiveＸコントロールで作成可能であるが，プログラム作成が容易なActiveＸコントロールを用いた。しかし制御速度が低速になる短所がある。

３．パラレル通信での調光器制御

調光器は，平成１３年度東海北陸地区国立大学等教室系技術職員合同研修「電気電子コース」の実習「赤外線リモコン送受信回路の製作とその応用回路」での応用回路として製作したものを，パソコンのプリンタポートから制御できるように改造した。

調光器の仕様は以下のとおりとした。

①パソコンからパラレル制御する。

②調光は１６段階とする。

③制御データは下位４ビットを使用し，調光する度にパソコンよりデータを送る。

④調光器パネルの調光表示は，調光器からの調光データを表示する。

パソコンのプリンタポートを汎用パラレルポートとして使用しているため，制御信号は適当に割り当てれば良いが，表１を参考に割り当てた。ただ，この信号を使用する時は，プリンタモニタが動作していると定期的に$\overline{\text{SLCT-IN}}$と$\overline{\text{AUTO FD XT}}$信号が発生するので，モニタを停止しておかなければならない。　パソコンのプリンタポートは，データ線の入出力を可能とする双方向フルセントロニクスモードに BIOS で設定し，そのアドレスマップを表２に示す。Portはポートアドレスで LPT1 の場合378h(プリンタポートのプロパティ)でデータポートとして，Port＋１はプリンタから送出するステータスポートとして，Port＋２はパソコンより送出する信号でコントロールポートとしてマップしている。プログラムでその信号に対応するビットに”０”を書き込むと，出力信号は”Ｈ”になるが，$\overline{\text{INIT}}$のみは”Ｌ”を出力する。

ピン番号	方向	記号名	備　考
1	I/O	$\overline{\text{STROBE}}$	データ取り込み同期信号
2	I/O	Data Bit 0	送受信データ・ビット0
3	I/O	Data Bit 1	送受信データ・ビット1
4	I/O	Data Bit 2	送受信データ・ビット2
5	I/O	Data Bit 3	送受信データ・ビット3
6	I/O	Data Bit 4	送受信データ・ビット4
7	I/O	Data Bit 5	送受信データ・ビット5
8	I/O	Data Bit 6	送受信データ・ビット6
9	I/O	Data Bit 7	送受信データ・ビット7
10	I	ACK	データ取り込み完了
11	I	BUSY	プリンタ受信不能
12	I	PE	ペーパ・エンド
13	I	$\overline{\text{SLCT}}$	プリンタ選択状態
14	O	$\overline{\text{AUTO FD XT}}$	印刷後自動改項
15	I	$\overline{\text{ERROR}}$	プリンタ・エラー
16	O	$\overline{\text{INIT}}$	プリンタ・初期化
17	O	SLCT IN	プリンタ・選択
18～25		Ground	

表１　プリンタポートの端子名

アドレス	ビット	信号名	方向	論理信号と信号レベルの対応
Port	0～7	データの入出力	I/O	0：“L”
Port＋1	2	割込みステータス	R	0：保留中の割込有
	3	ERROR	I	0：“L”
	4	SLCT	I	0：“L”
	5	PE	I	0：“L”
	6	~~ACK~~	I	0：“L”
	7	BUSY	I	1：“L”
Port＋2	0	$\overline{\text{STROBE}}$	O	0：“H”
	1	$\overline{\text{AUTO FD XT}}$	O	0：“H”
	2	$\overline{\text{INIT}}$	O	0：“L”
	3	SCLT IN	O	0：“H”
	4	割込み許可	W	0：不許可 1：許可
	5	ポート方向	W	0：出力 1：入力
＊ 方向 ＞ 1：入力，0：出力，W：書込み，R：読出し				

表２　プリンタポートのアドレスマップ

回路図を図２に，調光器パネルを図３に示す。操作はいたって簡単で，「点灯」クリックでランプが点灯可能になり，「消灯」クリックでランプが消灯する。「明」をクリックする毎に連続して明るくなり，「暗」で連続に暗くなり棒グラフと数値で明るさをモニタする。$\overline{\text{STB}}$ はパソコンからの調光データ下位４ビットを調光レジスタにセットする信号で，パネルの「明」をクリックするたびに調光データとともに送られる。調光レジスタのデータは，調光部でＤＡ変換されトライアックでランプを調光する。$\overline{\text{AUTO FD XT}}$信号は調光パネルの「点灯」「消灯」でデータを調光部に開閉す

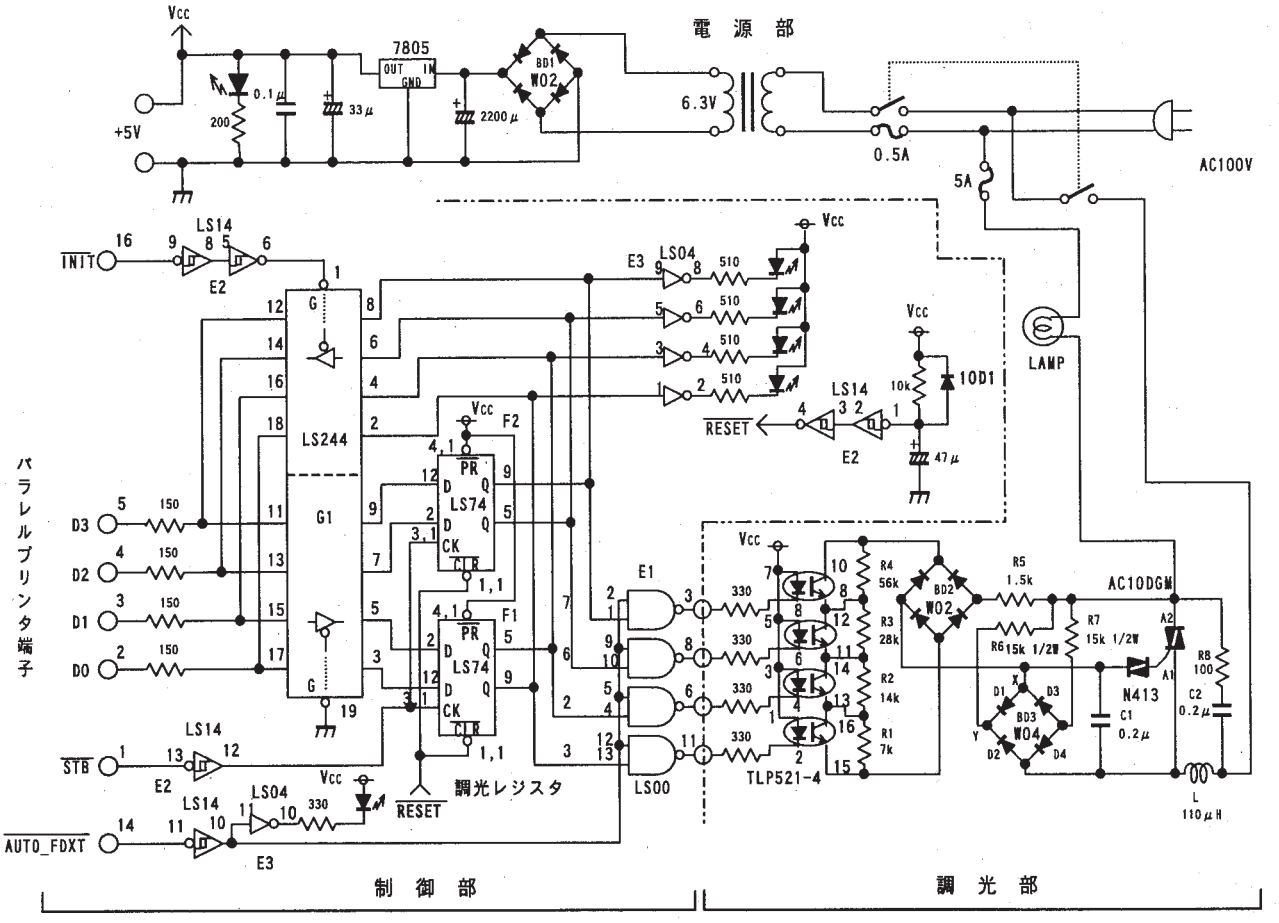

図2 パラレル制御調光器回路図

```
Dim Port%, Light%, Temp%
OutB Port, Light  ・・・・①
Temp = InpB(Port + 2)・・・②
Temp = Temp And &HDF  ・・・③
Temp = Temp Or &H01  ・・・④
OutB Port + 2, Temp  ・・・⑤
Temp = Temp And &HFE  ・・・⑥
Temp = Temp Or &H20  ・・・⑦
OutB Port + 2, Temp  ・・・⑧
```

リスト1　調光データ出力

る信号である。$\overline{\text{INIT}}$は調光レジスタのデータを乗せるゲート信号で，調光データのモニタに使用している。

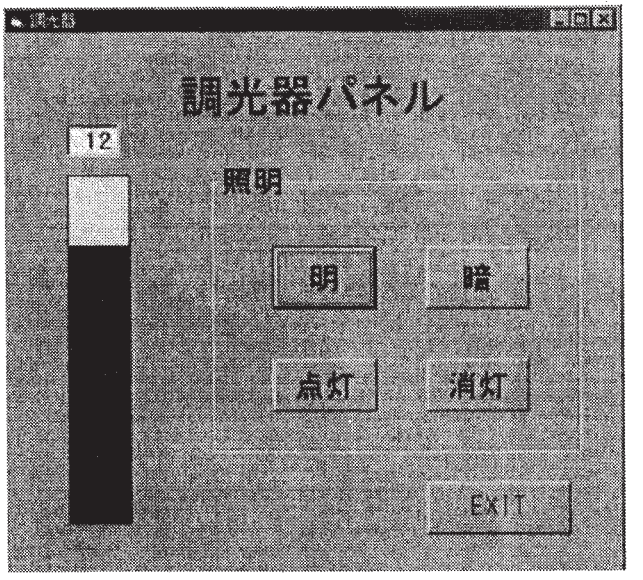

図3　調光器パネル

```
Declare Function InpB% Lib "Dio32S.DLL"(ByVal Port%)
Declare Sub OutB Lib "Dio32S.DLL"(ByVal Port%, ByVal DAT%)
```

リスト2　宣言文

リスト1は，ライブラリDIO32S.DLL を使用して調光データを出力するプログラムで，最初リスト2のように入出力命令がライブラリDio32S.DLLにあることを宣言しておく。

①で調光データを設定し，③でポート方向を”出力”に，④で$\overline{\text{STROBE}}$を”Ｌ”に設定し⑤で調光データ，$\overline{\text{STROBE}}$信号を出力しデータをレジスタにセットする。⑥⑦⑧はポート方向を”入力”に，$\overline{\text{STROBE}}$を”Ｈ”に戻す。

製作したインタフェースと流用した調光器を右の図４に示す。

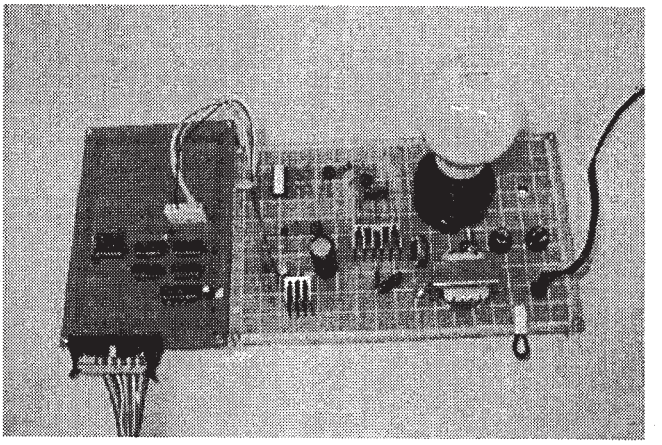

図４　インタフェースと調光器

４．シリアル通信での計測器制御とデータ取得

計測器として岩崎通信機製ロジックアナライザ ＳＬ４１２２ を使用した。アナライザは，サンプリングクロックでサンプルした１６チャンネルのデータをメモリ(４０９６ワード)に格納し，タイミング表示やステート表示をする。本研修では，シリアル通信によりアナライザを制御し，メモリデータを取得し，そしてタイミング表示する。

４－１．計測器制御

ＲＳ－２３２Ｃでのアナライザ制御の主な仕様を示す。

①通信条件：9600 ボー，８ビット長，ノンパリティ，１ストップビット
②転送コード：１６進コード　　③デリミタ：LF
④取得波形チャンネル：ｃｈ０～ｃｈ３　　⑤転送データ数：４０９６

[使用した制御コマンド]

使用したリモート制御コマンドとその意味を示す。

・ＫＹ　・・・　キーコマンドでアナライザのパネルのキースイッチを押した時と同じ動作
　　ＫＹ　Ｐ１　　Ｐ１：パネルキースイッチ４３個に割り振られた数字のキー
・ＪＸ　・・・　メモリからデータを読み出す
　　ＪＸ　Ｐ１，Ｐ２，Ｐ３，Ｐ４，Ｐ５
　　Ｐ１：転送コード指定　0 = Hex　1 = Bin　(Hexコード使用)
　　Ｐ２：メモリ指定　0 = Acq(取得)ﾒﾓﾘ　1 = Ref(参照)ﾒﾓﾘ　(Acqメモリ読取り)
　　Ｐ３：チャンネル指定　Ｐ３＝７　とするとHex転送の場合
　　bit3(MSB) = ch3, bit2 = ch2, bit1 = ch1, bit0 = ch0(LSB)は
　　に対応し，Ｄ(Hex データ)は，ch3 = 1, ch2 = 1, ch1 = 0, ch0 = 1 になる。
　　Ｐ４：転送開始番地
　　Ｐ５：転送データ数（最大２５６ワード）
・ＣＧ　・・・　Go to local　　リモートからローカル状態に戻す

取得チャンネルが４チャンネルでＪＸコマンド１回の最大データ転送数が２５６ワードのため，メモリデータ４０９６ワードの読出しは，ＪＸコマンドを１６回発行する。

４－２．MSComm コントロール

ＶＢで RS232C を利用するプログラムは，ActiveX コントロールとして付属している MSComm コントロール(MSCOMM32.OCXファイル)を使用して作成する。MSComm コントロールの初期化には，通信ポートの設定，通信条件の設定，ハンドシェイクの設定，送信時・受信時スレッショルドなどの設定などがあり，受信時スレッショルド以上のデータ量受信などにより割込みが生じると MSComm コントロールの OnComm イベントが生じる。

４－３．グラフ描画用ActiveXコントロール

アナライザから読み出されたメモリデータをタイミング表示するためにインターフェース社の「グラフ描画用ActiveXコントロール(VSA-0601)」を使用した。これはファイルIfxGraph2.ocxで提供されるカスタムコントロールである。本来はアナログ波形表示用であるが，デジタル波形表示として使用し，その他グリッド表示，ルーラ表示やマウスのイベントにも対応している。

４－４．ロジックアナライザ操作と波形表示

図５はＶＢで作成したロジックアナライザ　ＳＬ４１２２　の操作実行画面で，ボタン配置など実物器どおりとした。実物器に存在する任意のコマンドボタンをマウスでクリックすれば実物器の対応のキースイッチ押下と同じ動作をする。

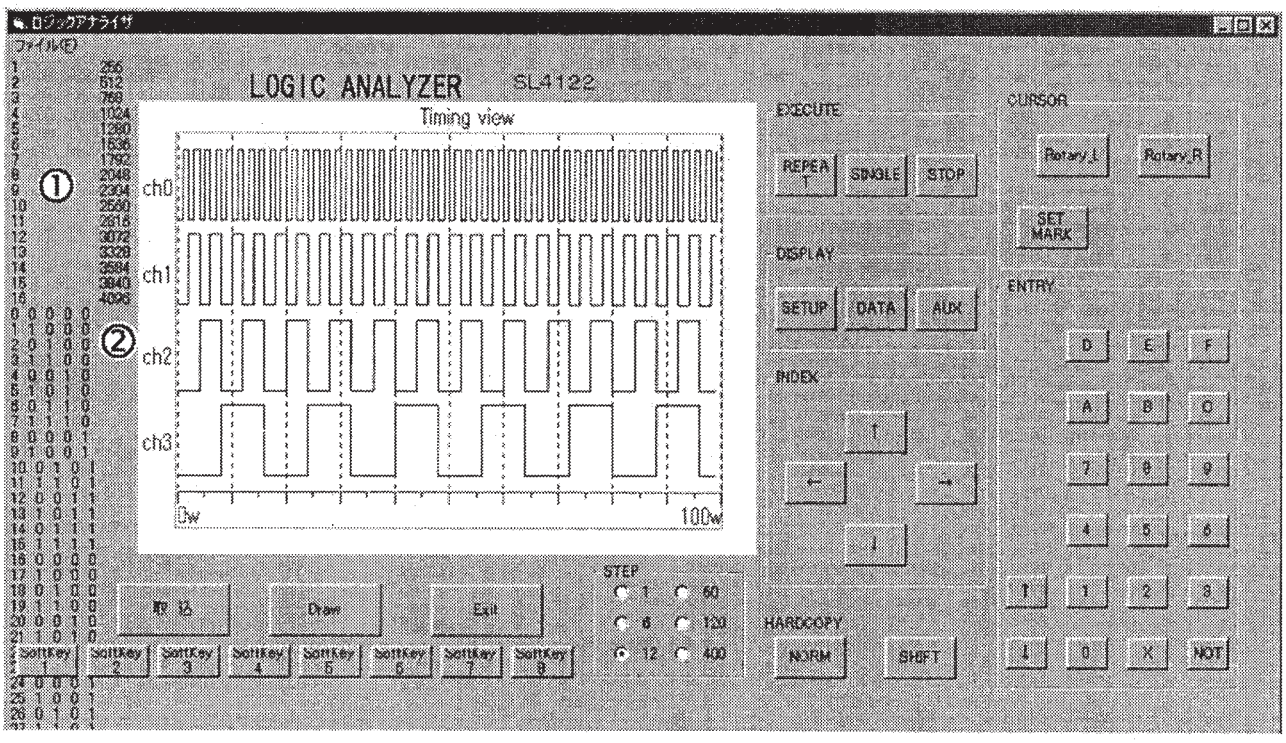

図５　ロジックアナライザ　ＳＬ４１２２　の操作実行画面

”取込”ボタンはアナライザから４０９６ワードのデータを１６回の転送により取得する(図５①)。図５②により読出した１ワードのデータをch0 ~ ch3に振り分けている。なお，①②は動作を確認するために便宜的に表示させている。

”Ｄｒａｗ”ボタンは読出したデータをタイミング表示する。波形表示は１００ワードであるが，他のワードの波形はマウスのホイールを回転して連続的に表示できる。波形表示は４０９６ワードまで

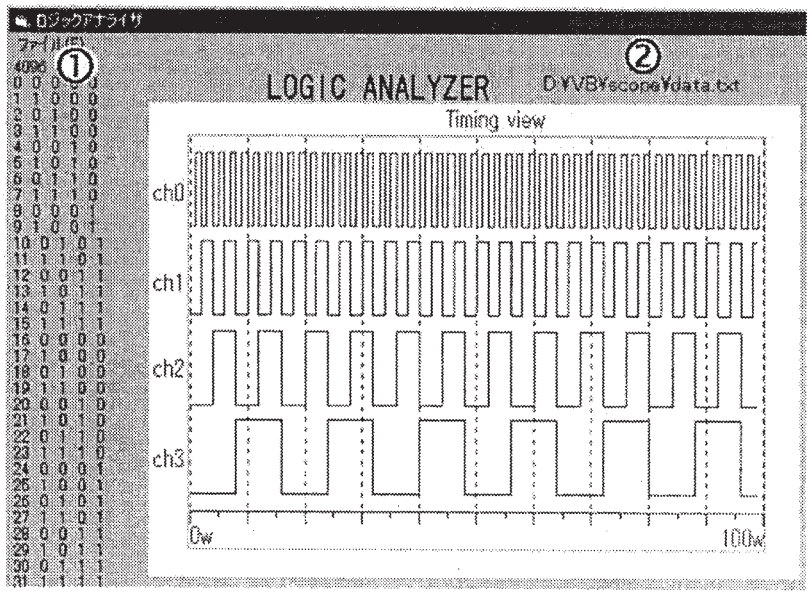

図６　ファイルからの読出し

表示可能で，オプションボタン STEP でホイールの１ステップ当たりワード数を変えることにより表示させたいところを設定することができる。

データ保存は，ファイルメニューの”保存”で任意のファイル名に保存する。

ファイルからのデータ読出しは，ファイルメニューの”開く”で行う。図６①で４０９６ワード読込まれたことを表し，②は読み出すファイルのディレクトリを表す。

５．まとめ

今回の研修において，Ｗｉｎｄｏｗｓ９８のＯＳのみではあるが，Ｗｉｎｄｏｗｓ上からプリンタポートより調光器制御を，ＲＳ－２３２Ｃポートよりロジックアナライザを制御し，データ取得，波形表示を行うインタフェースやプログラムを作成した。調光器制御ではインタフェースとプログラム作成することによりハードとソフトの関係が確認され，制御を行う醍醐味を若干味わえられたが，幾分お遊び的になってしまった。ロジックアナライザ制御ではタイミング表示のグラフ描画には自作の方法もあるが，安価で機能性のある既製のソフトを見つけ出し，使いこなすのも一つの技術であると思われる。

ＷｉｎｄｏｗｓでのＩ／Ｏ制御プログラム作成にＶＢを使用したが，Ｉ／Ｏ制御用ＤＬＬやActiveＸコントロールはＣ言語で作られており細部な制御やＯＳとの関わり合いなど不透明な部分があることや，ＡＰＩ使用での技術修得には至らなかったが，Ｗｉｎｄｏｗｓからの制御についての技術は大部分修得できた。

［参考文献］

１）渡辺明禎：ＶＢと製作で学ぶ初めてのパソコン応用工作　ＣＱ出版社

２）横山直隆：Visual Basic による計測・制御実験　ＫＫシータスク

３）金藤　仁：自動計測システムのためのＶＢ６入門　技術評論社

４）川口輝久他：かんたんプログラミング Visual Basic ６　基礎編　技術評論社

５）川口輝久他：かんたんプログラミング Visual Basic ６　応用編　技術評論社

６）ロジックアナライザ　ＳＬ４１２２　取扱説明書　岩崎通信機ＫＫ

７）シリアルデータアナライザインタフェース　取扱説明書　岩崎通信機ＫＫ

８）平成１３年度東海北陸地区国立学校等教室系技術合同研修電気電子コース
「赤外線リモコン送受信回路の製作とその応用回路」実習資料